

深入理解Java的内省与反射Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/644/2021\\_2022\\_\\_E6\\_B7\\_B1\\_E5\\_85\\_A5\\_E7\\_90\\_86\\_E8\\_c104\\_644998.htm](https://www.100test.com/kao_ti2020/644/2021_2022__E6_B7_B1_E5_85_A5_E7_90_86_E8_c104_644998.htm) 很多朋友在深入的接触Java语言后就会发现这样两个词：反射(Reflection)和内省(Introspector)，经常搞不清楚这到底是怎么回事，在什么场合下应用以及如何使用？反射相对而言，反射比内省更容易理解一点。用一句比较白的话来概括，反射就是让你可以通过名称来得到对象(类，属性，方法)的技术。例如我们可以通过类名来生成一个类的实例.知道了方法名，就可以调用这个方法.知道了属性名就可以访问这个属性的值，还是写两个例子让大家更直观的了解反射的使用方法：  
//通过类名来构造一个类的实例

```
Class cls_str=Class.forName("java.lang.String"). //上面这句很眼熟，因为使用过JDBC访问数据库的人都用过J
```

```
Object str=cls_str.newInstance(). //相当于String str=new String(). //通过方法名来调用一个方法 String methodName="length".
```

```
Method m=cls_str.getMethod(methodName,null).
```

```
System.out.println("length is " + m.invoke(str,null)). //相当
```

```
于System.out.println(str.length()). 上面的两个例子是比较常用方法。看到上面的例子就有人要发问了：为什么要这么麻烦呢?本来一条语句就完成了的事情干吗要整这么复杂?没错，在上面的例子中确实没有必要这么麻烦。不过你想想这样一个应用程序，它支持动态的功能扩展，也就是说程序不重新启动但是可以自动加载新的功能，这个功能使用一个具体类来表示。首先我们必须为这些功能定义一个接口类，然后我们
```

要求所有扩展的功能类必须实现我指定的接口，这个规定了应用程序和可扩展功能之间的接口规则，但是怎么动态加载呢?我们必须让应用程序知道要扩展的功能类的类名，比如是test.Func1，当我们把这个类名(字符串)告诉应用程序后，它就可以使用我们第一个例子的方法来加载并启用新的功能。这就是类的反射，请问你有别的选择吗? 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)